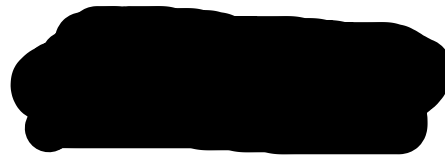


# Card counting

---



# Information

- 블랙잭 게임 소개
  - 딜러와 플레이어가 카드를 받아 21에 가까운 수를 만드는 사람이 이기며, 21을 초과하면 지는 게임.
  - 세계의 카지노에서 가장 널리 행해지는 플레이카드 게임.



# Playing card

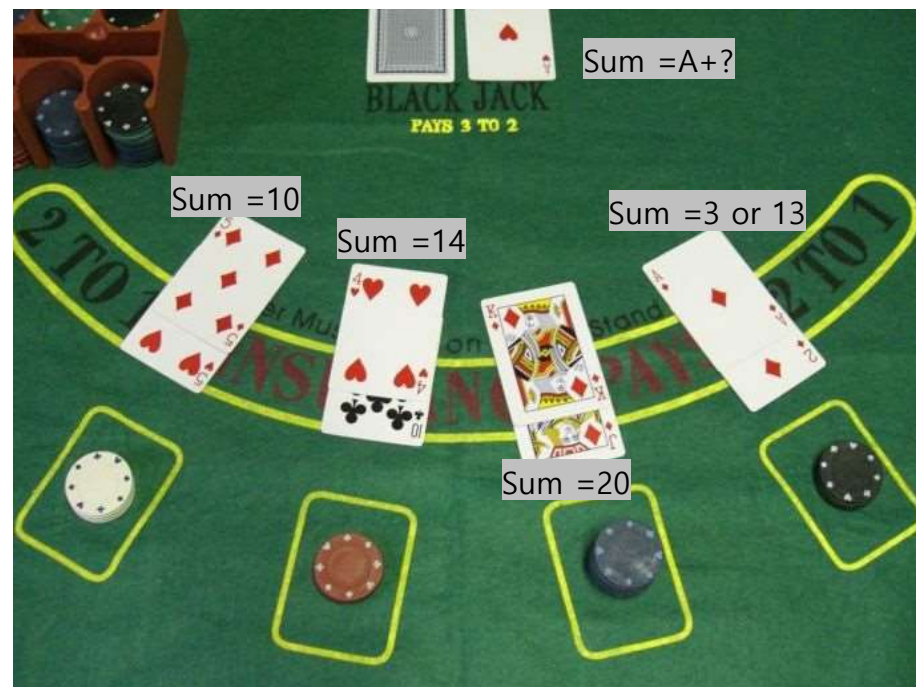
- 플레잉 카드 특징
  - 모양 : spade, heart, diamond, club로 4종류
  - 모양별로 A~K까지 13장
  - 한 묶음의 카드를 1 deck( $4 \times 13 = 52$ )라고 한다.



# Blackjack Rule

## ■ 기본 규칙

- 딜러와 플레이어가 카드를 받아 21에 가까운 수를 만드는 사람이 이기며, 21을 초과하면 지는 게임.
- 플레이어가 베팅하면 게임이 시작됨.
- 딜러가 자신을 포함한 플레이어들에게 카드 두 장을 나누어주는데, 딜러의 카드 한 장은 플레이어에게 보이지 않음.



# Blackjack Rule

## ■ 카드 deck 규칙

- 한 묶음의 카드를 1 deck( $4 \times 13 = 52$ 장).
- 일반적으로 카지노에서는 6 deck (=312장)을 사용.
- 6 deck의 80%를 사용할 때까지 k번 게임을 진행.
- 20%가 남으면, 그 게임을 마지막으로 플레이하고 다시 셔플링.



# Blackjack Rule

## ■ 기본 규칙 - 순서

- 카드 두 장을 받을 때까지 딜러와 플레이어는 번갈아가며 한 장씩 받음.
- 플레이어가 여럿일 경우, 플레이어는 시계방향으로 카드를 받음.
- 모두 두 장을 받으면, 플레이어가 먼저 게임을 진행.





# Blackjack Rule

- 플레이어 규칙
  - 딜러보다 먼저 플레이를 한다.
  - Hit : 카드 한 장을 더 받고 싶은 경우.
  - Stand : 카드를 더 이상 받고 싶지 않은 경우.

Dealer



Player



$$11+1=12$$

1 Hit



$$11+1+2=14$$

2 Hit



$$11+1+2+4=18$$

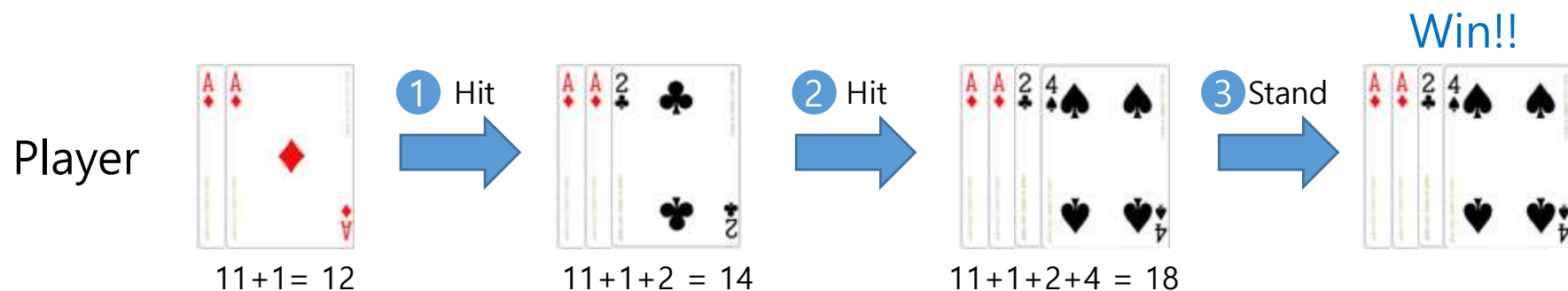
3 Stand



# Blackjack Rule

## ■ 딜러 규칙

- 플레이어가 모두 stand를 했을 경우, 딜러가 플레이할 차례.
- 딜러의 덮어둔 카드를 열고, 17 이상이면 stand, 16 이하면 hit 한다.





# Blackjack Rule

---

## ■ 승패 규칙




- 딜러와 숫자의 총합을 비교해서 높으면 승리, 낮으면 패배.
- Bust인 경우 패배(단, 딜러와 플레이어 모두 Bust인 경우 플레이어 패배.)
  - \* Bust : 숫자의 합이 21을 초과하는 경우
- 딜러의 카드와 합이 같으면 무승부.

## ■ 배팅 규칙

- 승리 시 : 배팅 금액의 2배를 돌려 받음
- 무승부 : 배팅 금액을 돌려 받음
- 블랙잭으로 승리 시 : 배팅금액의 2.5배를 돌려 받음
  - \* 블랙잭이란? 두 장의 카드만 받고 합이 21인 경우
- 패배 시 : 배팅 금액을 잃음

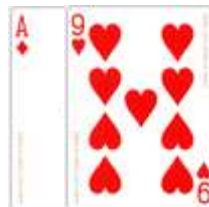

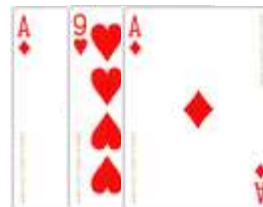
# Blackjack Rule

## ■ 카드 합산 방법 예시

$11 + 4 = 15$

$11 + 4 + 8 = 23(x)$   
 $1 + 4 + 8 = 13(o)$

$11 + 9 = 20$

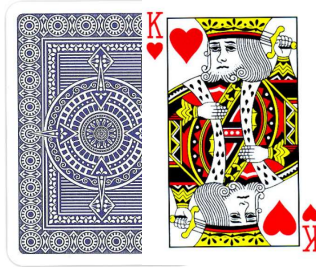
$11 + 9 + 1 = 21$

- 플레이어는 A(Ace)를 1과 11 중 자유롭게 사용 가능.
- 딜러는 A(Ace)를 11 로만 사용.

# Card counting

- 딜러의 숨겨진 카드는 무엇일까? (이해를 돕기 위한 1 deck example)

- 처음 받는 2장 중에 1장 이상 10, J, Q, K, A가 포함될 확률 :  $\frac{4*5}{52} = \text{약 } 38\%$



-> Player는 보이지 않는 딜러의 카드를 10이라고 가정.

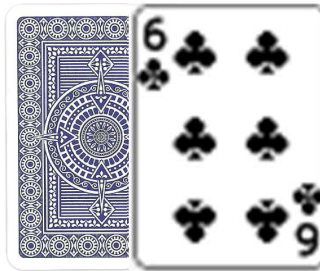
Why? 13장 중에 4장이 10(10, J, Q, K)이고, 10에 거의 근접한 8, 9, 11(Ace)까지 포함하면

절반 이상이기 때문. 10 근처로 카드를 받을 확률  $\frac{7}{13} = \text{약 } 53\%$ .

# Card counting

- 숫자를 세는 이유?

ex) 딜러가 6을 받을 경우



$$10(\text{가정}) + 6 = 16$$



딜러는 17 미만이면 무조건  
한 장을 더 받아야 함.



$$10(\text{가정}) + 6 + 10 = 26$$

남은 카드에 10이 많이 남아있다는 것을 안다면,  
딜러가 bust할 확률도 높다는 것을 알 수 있음.

$$\frac{\text{남은 10의 개수}}{\text{남은 카드}}$$

# Card counting

- 숫자를 세는 이유?
  - 남은 카드로 플레이어가 유리한지, 딜러가 유리한지에 대한 확률을 계산할 수 있음
  - 카드 카운팅을 통해 딜러의 버스트 확률이 높을 때 더 많이 배팅을 하거나, 플레이어가 불리 할 때 적게 배팅을 하는 등의 전략을 취할 수 있음.

ex) 한 판에 15장을 사용하였는데 숫자 10이 3장만 나왔었다고 해보자.  
10이 나올 확률은? (1 deck 사용)

$$\frac{4 * 4}{52} = 30.7\%$$

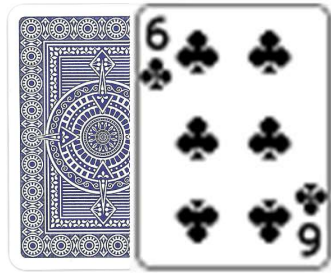
첫 판 시작 할 때

$$\frac{4 * 4 - 3}{52 - 15} = 35.1\%$$

다음 판 시작 할 때

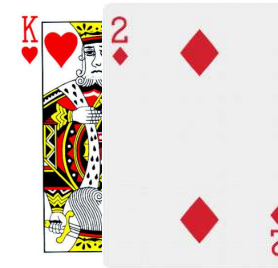
# Card counting

- Example (Hit, Stand), 6 decks



Dealer

$$10(\text{가정}) + 6 = 16$$



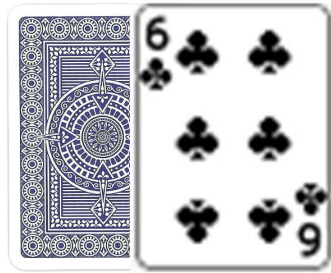
Player

Hit? Stand?



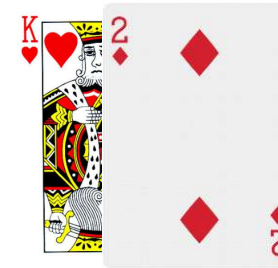
# Card counting

- Example (Hit, Stand), 6 decks



Dealer

$$10(\text{가정}) + 6 = 16$$



Player

Hit? Stand?

A	2	3	4	5	6	7	8	9	10	J	Q	K	전체
10	20	13	11	12	10	8	6	7	12	3	4	9	125

A~5이 나올 확률)

$$\frac{6 * 5 * 4 - 66}{312 - 125} = 28.8\%$$

딜러가 Bust할 확률)

$$\frac{6 * 8 * 4 - 59}{312 - 125} = 71.1\%$$

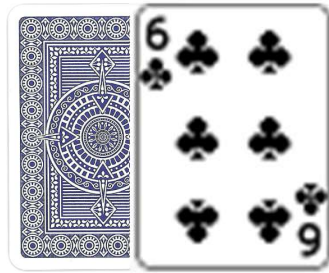
# Card counting strategy

- Card counting 전략

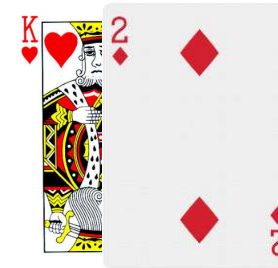
Card Strategy	2	3	4	5	6	7	8	8	10, J, Q, K	A	Level of Count
Hi-Lo	+1	+1	+1	+1	+1	0	0	0	-1	-1	+1
Hi-Opt I	0	+1	+1	+1	+1	0	0	0	-1	0	+1
Hi-Opt II	+1	+1	+2	+2	+1	+1	0	0	-2	0	+2
KO	+1	+1	+1	+1	+1	+1	0	0	-1	-1	+1
Omega II	+1	+1	+2	+2	+2	+1	0	-1	-2	0	+2
Red 7	+1	+1	+1	+1	+1	0 or +1	0	0	-1	-1	+1
Halves	+0.5	+1	+1	+1.5	+1	+0.5	0	-0.5	-1	-1	+3
Zen Count	+1	+1	+2	+2	+2	+1	0	0	-2	-1	+2

# High-Low Card counting

- Example (Hit, Stand), 6 decks



Dealer



Player

Hit? Stand?

A	2	3	4	5	6	7	8	9	10	J	Q	K	전체
10	20	13	11	12	10	8	6	7	12	3	4	9	125

A~5이 나올 확률)

$$\frac{6 * 5 * 4 - 66}{312 - 125} = 28.8\%$$

딜러가 Bust할 확률)

$$\frac{6 * 8 * 4 - 59}{312 - 125} = 71.1\%$$

Hi-Lo card counting)

$$\frac{+66 - 38}{3(deck)} = +9$$

# Card counting

## ■ Conclusion

- 실제로 사용되는 다양한 Card counting 전략이 있음.
- 위에서 소개한 Card counting 전략들은 확률적 계산을 기반으로 진행하지만, 확률 값을 근사화하고 단순화시킨 방법들이다.

$$\text{Ex) } \frac{+66 - 38}{3(\text{deck})} = +9$$

- Card counting 프로그램을 만들어서, 정확한 계산으로 카드를 세고 blackjack 게임에서 승률을 높이는 프로젝트를 진행해보자.

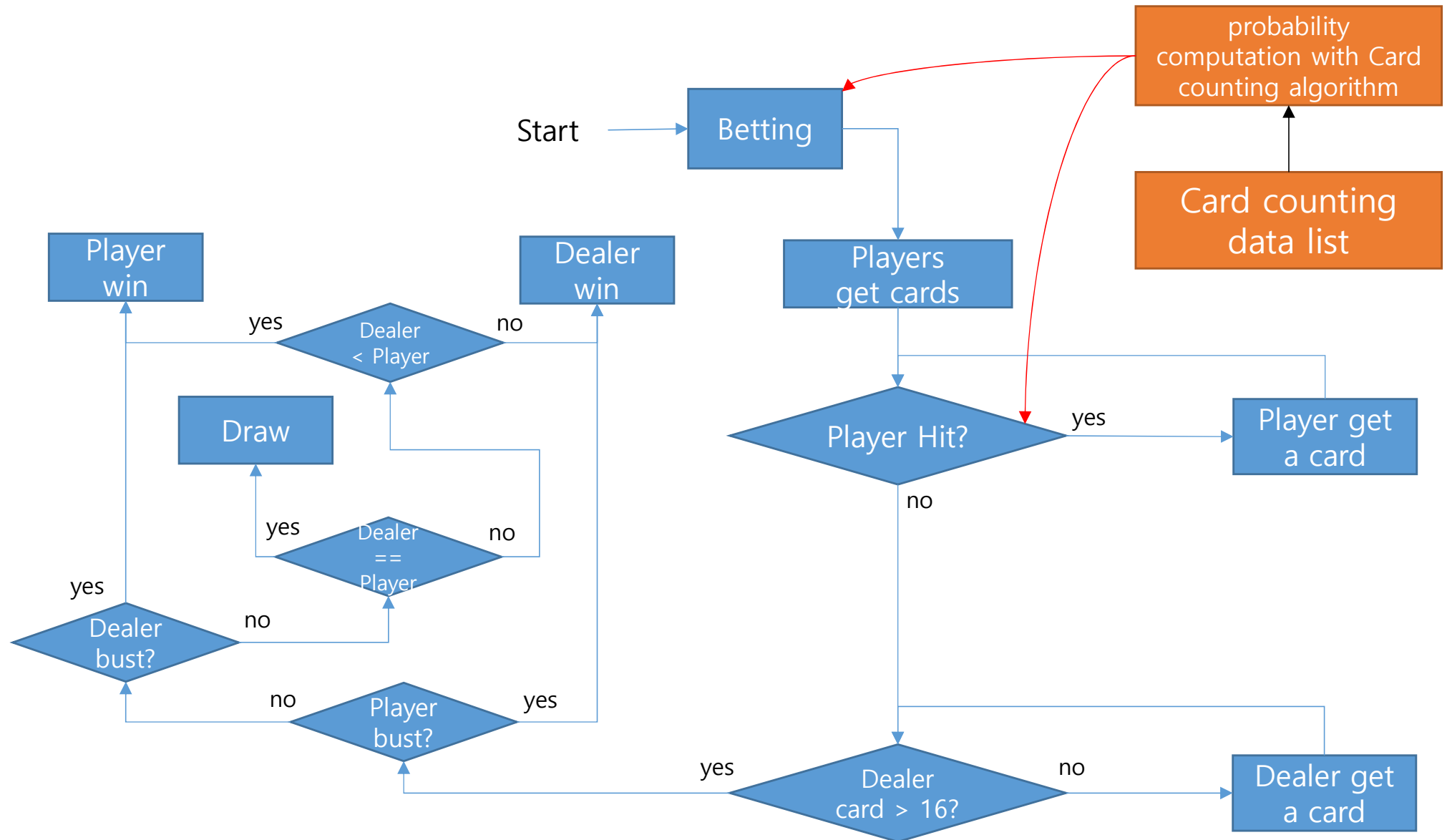
<card counting 정확한 계산>

$$\frac{6 * 8 * 4 - 59}{312 - 125} = 71.1\%$$

<Hi-Lo card counting>

$$\frac{+66 - 38}{3(\text{deck})} = +9$$

# Card counting Flowchart



# Card counting Project

1. Dealer, player 1, player 2가 같이 게임을 진행한다.
2. 플레이어들은 딜러의 카드 한 장을 볼 수 없다
3. 처음 두 장의 카드는 Dealer → player 1 → player 2 순서로 받는다.  
\* Dealer → player 1 → player 2 → Dealer → player 1 → player 2 (각 2장)
4. player 1은 카드 카운팅을 통한 확률 기반으로 Betting과 Hit, Stand를 결정 할 수 있다.
5. player 2는 딜러와 마찬가지로 17이상이면 Stand, 미만이면 Hit을 한다.
6. 게임을 n번 진행 후, player 1과 player 2의 승률을 계산 후 비교한다.





# Card counting Project

---

- Conditions

- 카드는 3 deck(156)으로 진행하며, 카드의 80% 이상 사용한다면 다음 게임을 시작할 때 모든 카드(사용한 카드 포함)를 다시 섞어준다.
- 초기 자금은 100000원이다.
- Player 1과 Player 2는 각각 게임 시작 시 100~1000원, 500원의 금액을 베팅한다.

# Card counting Project

---

- Function implementation

- Shuffle() : 카드를 섞어주는 기능을 한다. 6 deck의 80% 이상 사용하면, 다음 게임을 시작할 때, 모든 카드(사용한 카드 포함)를 다시 섞어준다.
- HitStatus() : 카드를 Hit할지, Stand할지 정해주는 기능을 한다. Cardcounting을 고려한 HitStatus(), dealer와 simple player의 HitStatus()를 고려해서 작성한다.
- SumofNum() : 현재 플레이어의 카드 숫자 합이 얼마인지 계산하는 기능을 한다.
- 이외에 필요한 기능을 자유롭게 작성한다.

# 결과 예시

- 10만 번 진행

```
Play Game : 100000
<< Player1 >>
승리 : 42094 무승부 및 패배 : 57906
money : -536400

<< Player2 >>
승리 : 40614 무승부 및 패배 : 59386
money : -572800
```

$$* \text{승률} : \frac{\text{승리 수}}{\text{게임 수}} * 100(\%)$$

- 만 번 진행

```
Play Game : 10000
<< Player1 >>
승리 : 4245 무승부 및 패배 : 5755
money : 48250

<< Player2 >>
승리 : 4018 무승부 및 패배 : 5982
money : 27350
```

- 100번 진행

```
Play Game : 100
<< Player1 >>
승리 : 42 무승부 및 패배 : 58
money : 99550

<< Player2 >>
승리 : 38 무승부 및 패배 : 62
money : 98350
```

# Report

---

- 보고서 + 소스코드(주석 포함)를 압축하여,  
`Assignment3_학번_이름.zip`으로 [KLAS]-[과제 제출]에 제출
- 보고서는 ms word 또는 한컴오피스 한글로 작성
- 소스코드는 .c, .cpp, .h 파일만 제출

Due date: 05/24 23:59:59까지

# Report - details

---

- **과제 설명**
- **접근 방법**
  - 카드 카운팅을 어떻게 적용했는지 설명
  - 과제를 진행하면서 생겼던 이슈는 어떻게 해결하였는지 설명
- **실험 결과**
  - 결과 화면 캡처 및 분석
- **고찰**
  - 과제 진행 과정 및 결과에 대해 자신의 고찰을 작성

# END OF PRESENTATION

---

Q&A